

Relationale Datenbanken als RDF-Speicher

Rows vs. Columns

- Tripel-Table
- Property-Table
- Vertical Partitioning

Tabellenstrukturen für RDF

Triple-Table

| subject | predicate | object |
|---------|-----------|--------|
| s1 | a1 | o1 |
| s1 | a2 | o2 |
| s2 | a1 | o1 |
| s3 | a1 | o3 |
| s3 | a2 | o4 |
| s3 | a3 | o5 |
| . | . | . |
| . | . | . |
| . | . | . |

Property Table - single class

Class A

| subject | a1 | a2 | a3 | ... | an |
|---------|----|------|------|-----|-----|
| s1 | o1 | o2 | null | ... | ... |
| s2 | o1 | null | null | ... | ... |
| s3 | o3 | o4 | o5 | ... | ... |
| . | . | . | . | . | . |
| . | . | . | . | . | . |

Class X

| subject | x1 | x2 | x3 | ... | xm |
|---------|----|------|------|-----|-----|
| s4 | o6 | o11 | o2 | ... | ... |
| s9 | o3 | o13 | null | ... | ... |
| s11 | o7 | null | o1 | ... | ... |
| . | . | . | . | . | . |
| . | . | . | . | . | . |

Vertical Partitioning

Property a1

| subject | object |
|---------|--------|
| s1 | o1 |
| s2 | o1 |
| s3 | o3 |

Property a2

| subject | object |
|---------|--------|
| s1 | o2 |
| s3 | o4 |

Property a3

| subject | object |
|---------|--------|
| s3 | o5 |

Property an

| subject | object |
|---------|--------|
| ... | ... |

Property Table - multiple types

| subject | type | b1 | b2 | y1 | y2 | ... |
|---------|------|------|------|------|----|-----|
| s1 | B | o1 | o2 | null | o3 | ... |
| s2 | B | o1 | null | null | o8 | ... |
| s3 | Y | null | o4 | o5 | o2 | ... |
| s5 | Y | null | null | o8 | o7 | ... |
| . | . | . | . | . | . | . |
| . | . | . | . | . | . | . |

Example 1a: In which years have database books been published?

Query in SPARQL over RDF-Graph

```
SELECT ?year WHERE {  
  ?book type Book .  
  ?book title "DBMS" .  
  ?book issued ?year  
}
```

Query in SQL over Triple-Table

```
SELECT T3.object AS year FROM Triple-Table T1,  
      Triple-Table T2,  
      Triple-Table T3  
WHERE  
  T1.predicate = 'type' AND T2.predicate = 'title' AND  
  T3.predicate = 'issued' AND  
  T1.object = 'Book' AND T2.object = 'DBMS' AND  
  T1.subject = T2.subject AND T2.subject = T3.subject
```



Example 1b In which years have database books been published?

Query in SPARQL over RDF-Graph

```
SELECT ?year WHERE {  
  ?book type Book .  
  ?book title "DBMS" .  
  ?book issued ?year  
}
```

Query in SQL over Property-Table

```
SELECT year FROM Property-Table WHERE  
  type = 'Book' AND title = 'DBMS'
```



Example 1c In which years have database books been published?

Query in SPARQL over RDF-Graph

```
SELECT ?year WHERE {
  ?book type Book .
  ?book title "DBMS" .
  ?book issued ?year
}
```

Query in SQL over Vertical Partitioning

```
SELECT Is.object AS year FROM Type Ty,
       Title Ti,
       Issued Is
WHERE
  Ty.object = 'Book' AND
  Ti.object = 'DBMS' AND
  Ty.subject = Ti.subject AND Ti.subject = Is.subject
```



Example 2: In which predicates do persons appear as objects?

Query in SPARQL over RDF-Graph

```
SELECT ?predicate WHERE {
  ?person rdf:type foaf:Person .
  ?subject ?predicate ?person
}
```

Query in SQL over Triple-Table

```
SELECT T2.predicate FROM Triples T1,
       Triples T2
WHERE
  T1.predicate='Type' AND T1.object='Person' AND
  T2.object = T1.subject
```

Property-Table (single class) and Vertical Partitioning are going to fail.

